

PATENT ABSTRACTS OF JAPAN

(11) Publication number: **10326197 A**(43) Date of publication of application: **08.12.98**

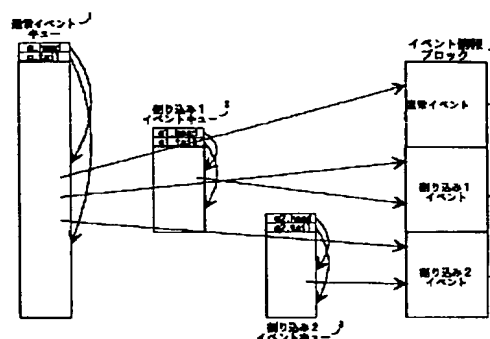
(51) Int. Cl.

G06F 9/46
G06F 13/00(21) Application number: **09136899**(71) Applicant: **SHARP CORP**(22) Date of filing: **27.05.97**(72) Inventor: **AIDA WATARU****(54) QUEUE PROCESSING SYSTEM****(57) Abstract:**

PROBLEM TO BE SOLVED: To accelerate queue processing by respective events in an input queue to successively process without performing any interruption inhibition and making the input queue processing possible, in the case of successive processing in the system of electronic computer or the like, which is provided with communication equipment or a communication function, for temporarily receiving respective events and processing them in order.

SOLUTION: Dedicated queues 1, 2 and 3 are prepared for every event and an event information block area 4 is acquired from a common memory area. Besides, the ordinary event queue 1 can use an event based on a timer or hardware interruption as an ordinary event based on the software by preparing all the event information block areas.

COPYRIGHT: (C)1998,JPO



This Page Blank (uspto)

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-326197

(43) 公開日 平成10年(1998)12月8日

(51) Int.Cl.⁶

G 0 6 F 9/46
13/00

識別記号

3 4 0
3 5 3

F I

G 0 6 F 9/46
13/00

3 4 0 C
3 5 3 P

審査請求 未請求 請求項の数 6 O L (全 8 頁)

(21) 出願番号

特願平9-136899

(22) 出願日

平成9年(1997)5月27日

(71) 出願人 000005049

シャープ株式会社

大阪府大阪市阿倍野区長池町22番22号

(72) 発明者 合田 亙

大阪府大阪市阿倍野区長池町22番22号 シ
ャープ株式会社内

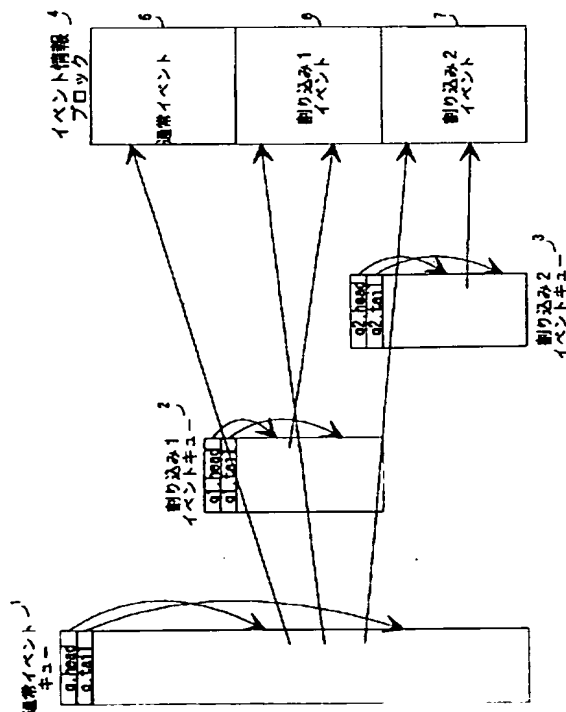
(74) 代理人 弁理士 梅田 勝

(54) 【発明の名称】 キュー処理方式

(57) 【要約】

【課題】 通信機器、通信機能を具備した電子計算機等の、各イベントを随時受け付け、それらを順番に処理するシステムにおいて、各イベントを入力キューに登録し、順次処理する際に、割り込み禁止を行わずに、入力キューの処理を可能とすることにより、キュー処理の高速化を提供する。

【解決手段】 各イベント毎に専用のキュー1、2、3を用意し、イベント情報ブロック領域4を共通のメモリ領域から取得する。また、通常イベントキュー1は全てのイベント情報ブロック領域数を用意することにより、タイマーやハードウェア割り込みによるイベントをソフトウェアによる通常イベントとしての転用を可能とする。



【特許請求の範囲】

【請求項1】 ソフトウェアからの通常イベントと、タイマー、ハードウェア割り込み等によるイベントを入力とするとともに、これら各イベントを随時受け付け、それらを順番に処理するシステムにおけるキュー処理方式であって、

割り込みによりキュー操作が発生する場合、割り込みにより発生するイベントを専用のキューに置くことにより、キュー操作時に割り込みを禁止せずにイベントを処理することを特徴とするキュー処理方式。

【請求項2】 複数の割り込みの種類があり、その内の1種類の割り込みイベントに関するキューの処理中に他の種類の割り込みが発生する場合、それらを別のキューで処理することを特徴とする請求項1記載のキュー処理方式。

【請求項3】 通信ハードウェアからの割り込み専用のキューとタイマー割り込みによる割り込み専用のキューの2つを別に設けることを特徴とする請求項1または2記載のキュー処理方式。

【請求項4】 それぞれのイベントを共通のメモリ領域に保持することを特徴とする請求項1または2または3記載のキュー処理方式。

【請求項5】 それぞれのイベントが同一のフォーマットに従うとともに、イベントが同一のフォーマットであることを利用して、各キューの処理を同一の方式で処理することにより処理方法を簡単化したことを特徴とする請求項1ないし4のいずれかに記載のキュー処理方式。

【請求項6】 別々のキューに入れられたイベントを処理後、別のイベントキューに移して再利用することを特徴とした請求項1ないし5のいずれかに記載のキュー処理方式。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 本発明は、ソフトウェアからの通常イベントと、タイマー、ハードウェア割り込み等によるイベントとを入力とするシステム、例えば、通信機器、電子計算機等の、メモリー等のデータをキュー構造により管理するシステムにおけるキュー処理方式に関する。

【0002】

【従来技術】 図4は、メモリー等のデータ構造に一般的に用いられているキュー構造の概念図を示している。あるデータを追加するときは、最後のデータであるDATA#Nの最後に追加し、キューに保持されているデータを取り出すときには、キューの先頭のデータDATA#Mから取り出す。したがって、キューの先頭と末尾を示す、二つのポインタを用いることによってキューを実現することができる。

【0003】 実際の実現方法としては、各データ領域に隣接するデータ領域のリンク情報を保持する方法が良く

用いられる。この様子を図5に示す。例えば、図5のキューにあるデータを保持する場合、末尾のデータであるDATA#Nのリンク情報をDATA#Nへリンクされているように書き換え、DATA#N+1のリンク情報をDATA#Nにセットし、DATA#Nへのポインタがセットされているq. tailをDATA#N+1へのポインタに書き換える。逆に、キューに保持されているデータを取り出す場合、DATA#M+1のリンク情報をクリアし、現在DATA#Mへのポインタがセットされているq. headをDATA#M+1に書き換える。

【0004】 ここで、上記のキュー処理の途中で割り込みが発生する場合、キューの整合性を保持するため、一定のキュー操作の間は割り込みを禁止する必要がある。図5のキュー構造に対し、割り込みを禁止せずに、キュー操作中に割り込みが発生した場合の様子を図6に示す。

【0005】 図6(a)のように、DATA#MからDATA#Nまでのデータがキューに貯えられているところで、新たなデータDATA#N+1をこのキューに追加する。そのために、まず末尾のデータであるDATA#NとDATA#N+1のリンク情報をそれぞれ図6(b)のようにセットする。

【0006】 この時点で、DATA#N+1'をこのキューに追加するような割り込み処理が発生したとすると、q. tailはDATA#Nを指しているため、図6(c)のようにDATA#Nのリンク情報をDATA#N+1'を指し示すようにセットし、さらにq. tailをDATA#N+1'へのポインタに書き換える。

【0007】 この時点で割り込み処理が終了し、DATA#N+1をキューに追加する処理が戻り、最終的にq. tailのポインタをDATA#N+1にセットする。しかしながら、このような処理では、図6(d)のような状態になり明らかに矛盾が生じていることが分かる。

【0008】 したがって、図5のような構成をとる場合、キュー操作中は割り込み禁止とする必要がある。このため、特開平7-13782号公報に開示された発明では、従来必要であった割り込み禁止区間を分割することにより、個々の割り込み禁止時間を一定の時間に押さえ、かつ矛盾なくキュー操作を行うことを可能としている。

【0009】

【発明が解決しようとする課題】 しかしながら、キューの整合性を保持するために割り込み禁止区間を設ける場合、詳細な論理関係の検討、タイミングの検討が必要であり、複数の割り込みの間で相互作用がある場合は、さらに高度な技術が要求される。

【0010】 さらに、割り込み禁止区間の設定が適当かどうか動作確認を行う場合、割り込みのタイミングを制

御することや、複数の割り込みの間で相互作用があるときには割り込み処理中に別の割り込みを発生させること等が必要となり、それらの制御は非常に困難である。

【0011】また、通信機器、通信機能を具備した電子計算機に应用する場合、高速の通信を行うために、割り込みにより通信処理が行われる。高速の通信では、上記の課題に加え、一定時間内に処理を行う必要があり、割り込み禁止により処理時間の遅延が発生するのは望ましくなく、できるだけ簡素な処理を行う必要がある。本発明は、上記課題に鑑みてなしたものであり、上記課題を解決したキュー処理方式を提供するものである。

【0012】

【課題を解決するための手段】請求項1のキュー処理方式は、ソフトウェアからの通常イベントと、タイマー、ハードウェア割り込み等によるイベントを入力するとともに、これら各イベントを随時受け付け、それらを順番に処理するシステムにおけるキュー処理方式であつて、割り込みによりキュー操作が発生する場合、割り込みにより発生するイベントを専用のキューに置くことにより、キュー操作時に割り込みを禁止せずにイベントを処理するものである。

【0013】請求項2のキュー処理方式は、請求項1記載のキュー処理方式において、複数の割り込みの種類があり、その内の1種類の割り込みイベントに関するキューの処理中に他の種類の割り込みが発生する場合、それらを別のキューで処理することを特徴とするものである。

【0014】請求項3のキュー処理方式は、請求項1または2記載のキュー処理方式において、通信ハードウェアからの割り込み専用のキューとタイマー割り込みによる割り込み専用のキューの2つを別に設けることを特徴とするものである。

【0015】請求項4のキュー処理方式は、請求項1または2または3記載のキュー処理方式において、それぞれのイベントを共通のメモリ領域に保持することを特徴とするものである。

【0016】請求項5のキュー処理方式は、請求項1ないし4のいずれかに記載のキュー処理方式において、それぞれのイベントが同一のフォーマットに従うことを特徴とし、イベントが同一のフォーマットであることを利用して、各キューの処理を同一の方式で処理することにより処理方法を単純化したことを特徴とするものである。

【0017】請求項6のキュー処理方式は、請求項1ないし5のいずれかに記載のキュー処理方式において、別々のキューに入れられたイベントを処理後、別のイベントキューに移して再利用することを特徴としたものである。

【0018】

【発明の実施の形態】以下、本発明の実施の形態につい

て説明する。図1は、本発明キュー処理方式におけるキューの構成方法を示した図、図2は、本発明キュー処理方式におけるイベントのフォーマット図、図3は本発明キュー処理方式の具体例を示したブロック図である。

【0019】まず、図1及び図2を用いて、本発明の概要を説明する。本発明の主たる構成を図1に、また、イベント情報ブロックの個々のイベントの内部構成を図2に示す。イベント情報ブロック4はイベントの内容を示すデータを保持する。イベント情報ブロック4は、連続したメモリ領域であり、通常イベント情報ブロック領域5と各種の割り込みに対するイベント毎にイベント情報ブロック領域6、7を持つ。

【0020】イベントの内容は、図2に示すように、どの処理部へのイベントかを示す送信先、どの処理部からのイベントであるかを示す送信元、イベントの種類を表すイベント種別、パラメータ長、イベント種別に対するパラメータから構成される。これは、通常イベントとその他の割り込みイベント全てについて同様の構成をとる。

【0021】通常イベントと上記の各種の割り込みに対するイベントに対し、それぞれのイベントキュー1、2、3を持つ。各イベントキュー1、2、3はキューエントリの先頭へのポインタであるq. headと最後尾へのポインタであるq. tailを持ち、イベントキューの各エントリにイベント情報ブロック4へのポインタを保持する。

【0022】各種割り込みイベントについて、イベント情報ブロック4で用意できるエントリー数だけ、それぞれのイベントキュー2、3にエントリーを用意する。通常イベントキュー1は、通常イベント情報ブロック5と各割り込みイベント情報ブロック6、7のエントリー数の合計数のキューエントリーを用意する。以上の構成により、割り込みによらないイベントの場合、イベントのデータを通常イベント情報ブロック領域5に保持し、通常イベントキュー1に追加する。

【0023】それに対し、割り込み1イベントが発生した場合は、割り込み1イベント情報ブロック領域6にイベントの内容を保持し、割り込み1イベントキュー2にイベントキューを追加する。割り込み2イベントが発生した場合も同様である。複数のイベント処理部が存在する場合等、割り込みイベントキュー2、3をあるイベント処理部から別のイベント処理部に渡すときは、通常イベントとして渡す。

【0024】その際、通常イベント、割り込みイベントのエントリー数全ての個数だけ、通常イベントキューがあるため、割り込みイベント情報ブロック領域6、7に保持されているデータを通常イベント情報ブロック領域5に移動することなく、割り込みイベントキューの登録除外し、通常イベントキュー1に割り込みイベント情報ブロック領域6、7に保持されているエントリーへのポイン

タを再登録する。

【0025】次に、本発明の具体的な実施例を図3を参照して説明する。通信機器において、通常メッセージを受信した場合、割り込みによって通知されるのが一般的である。また、通信相手にメッセージを送信した場合、一定時間内にそのメッセージに対する反応を期待することがある。そのために、通信相手にメッセージを送信した直後にタイマーをセットし、一定時間経過後に確認メッセージを受信しないと、タイマー割り込みによって通知され、再送制御が開始されるのが一般的である。

【0026】上記より本実施例では、通常イベント以外の割り込みイベントとして受信割り込みイベントとタイマーイベントの2つを想定し、本発明キュー処理方式のキュー構造を示した図1においては、割り込み1イベントを受信イベント、割り込み2イベントをタイマーイベントとする。

【0027】イベントキュー11は本発明キュー処理方式のキュー構造をとる。スケジューラ12は、イベントキュー11に登録されたイベントの内容を解析し、どのイベント処理部13、14、15へのイベントかを判別する。その結果を元に対応するイベント処理部にデータを渡し、制御権を移す。

【0028】また、イベント処理部等で発生した通常イベント8とタイマーイベント9、受信イベント10のイベントキュー11への登録も行う。イベント処理部には、レイヤ3イベント処理部13、レイヤ2イベント処理部14、レイヤ1イベント処理部15がある。各処理部にて、イベントに対する各プロトコルデータを生成する。

【0029】通信機器においては、OS1リファレンスモデルにてレイヤ1からレイヤ7まで層毎に各プロトコルが規定されている。本実施例においてもその概念に基づき、層毎にプロトコルを実現しているが、図3におけるレイヤ1、レイヤ2、レイヤ3はOS1リファレンスモデルのそれを表しているのではなく、各層の上下関係のみを概念的に表すものである。

【0030】したがって、送信時はレイヤ3イベント処理部13で生成されたプロトコルデータをレイヤ2イベント処理部14に渡し、レイヤ2イベント処理部14で生成されたプロトコルデータをレイヤ1イベント処理部15に渡し、レイヤ1イベント処理部15で生成されたプロトコルデータを送受信モジュール16に渡し、送信を行うことになる。受信時はこの逆の手順を踏むことになる。

【0031】送受信処理モジュール16にて、データの送受信処理が行われる。まず、送信手順について説明する。上位層からの送信要求は、図3のレイヤ3プリミティブ8の中の一つに定義されている。この送信要求が発生すると、スケジューラ12は通常イベント情報ブロック領域5より、空きブロックを検索し、このイベントに

付随するパラメータを含めて、通常イベント情報ブロック5に登録し、必要に応じてq. headおよびq. tailを書き換え、通常イベントキュー1の最後のエントリーの次に先ほどの通常イベントキューに登録する。

【0032】スケジューラ12は、一定周期で通常イベントキュー1、受信イベントキュー（割り込み1イベントキュー）2、タイマーイベントキュー（割り込み2イベントキュー）3にイベントが登録されているかどうかをチェックする。レイヤ3イベント処理部13へ入力されるべき通常イベントがあることが分かるので、通常イベントキュー1より登録を除外し、イベントを解析し、レイヤ3イベント処理部13にイベントを渡す。

【0033】レイヤ3イベント処理部13において、入力イベントからプロトコルデータを生成し、レイヤ2イベント処理部14に渡す。そのために、レイヤ3イベント処理部13がスケジューラ12のイベント登録ルーチンをコールし、プロトコルデータを通常イベントキュー1に登録し、処理権をスケジューラに渡す。スケジューラ12は、レイヤ2イベント処理部14へ入力されるべきイベントが存在するので、通常イベントキュー1より登録除外し、レイヤ2イベント処理部14に制御権を渡す。

【0034】同様な処理を行い、レイヤ2イベント処理部14で生成されたプロトコルデータはスケジューラを介してレイヤ1イベント処理部15に渡す。レイヤ1イベント処理部15では、レイヤ1プロトコルデータを生成し、送受信処理モジュール16に渡し、送信処理が行われる。

【0035】各レイヤ処理モジュールでは、一般に送信先アドレス、送信元アドレス、データ長等の情報を含むヘッダもしくはトレーラを、上位層から受け付けた送信データに付加するのが主な処理になる。したがって、レイヤ3プリミティブにて送信要求イベントを受け付ける時に、下位層にて付加されるヘッダもしくはトレーラのサイズを見越して通常イベント情報ブロックを割り当てる。

【0036】そして、レイヤ3イベント処理部13では、イベント情報ブロックに直接、ヘッダもしくはトレーラを付加し、レイヤ2イベント処理部14にイベント渡すときには、同じイベント情報ブロックを通常イベントキュー1に登録する。レイヤ2処理部14からレイヤ1処理部15にプロトコルデータを渡すときも同様である。このように、イベント情報ブロックを再利用することにより、送信データの無駄なコピーを防ぐことができる。

【0037】次に、受信手順の説明を行う。送受信モジュール16でデータを受信すると、割り込みによりスケジューラ12に通知される。スケジューラ12では、空き受信イベント情報ブロックを検索し、受信データを保持し、受信イベントキュー（割り込み1イベントキュー

一) 2に登録する。スケジューラ12が制御権を得ると、イベントキューをチェックし、その結果、受信イベントキュー2にイベントが登録されていることが分かる。その後、受信イベントを解析する。

【0038】通常イベント、受信イベント、タイマーイベントとともに同じ構成であるため、通常イベントの解析処理と全く同じ処理を行えばよい。解析の結果、レイヤ1イベント処理部15へのイベントであることが分かるので、そのイベントを受信イベントキュー2より登録除外し、レイヤ1イベント処理部15に受信イベントを渡す。

【0039】レイヤ1イベント処理部15では、受信データにエラーが生じていなければレイヤ1プロトコルを実現するためのヘッダもしくはトレーラを削除して、レイヤ2イベント処理部14に渡すために、通常イベントキュー1に登録する。この際、新たに通常イベント情報ブロックを取得しレイヤ2イベント処理部14に渡すデータを生成せずに、渡された受信イベント情報ブロックをそのまま利用し、レイヤ2イベント処理部14にデータを渡すことにより、受信データのコピーの処理を大幅に激減できる。レイヤ2イベント処理部14においても同様に受信イベント情報ブロック領域で、レイヤ3イベント処理部13へ渡すデータを生成し、通常イベントキューに登録する。

【0040】タイマーイベントが発生すると、割り込みによりスケジューラ12に通知される。スケジューラ12では空きタイマーイベント情報ブロックを検索し、必要な情報をイベント情報ブロックに登録する。スケジューラ12が制御権を得ると、イベントキューをチェックすることにより、タイマーイベントキュー(割り込み2イベントキュー)3にイベントが登録されていることが分かるので、該当するイベント処理部にイベントを渡す。ここでも、各イベント情報ブロックのフォーマットは統一されているため、独自の処理を行う必要はない。

【0041】

【発明の効果】以上に説明したとおり、割り込み処理毎

にイベントキューを用意することにより、割り込み禁止処理を行わず、さらに複数の割り込み処理の間で互いに干渉を受けずに、キューの処理が可能となる。したがって、割り込みによる処理の遅延が発生せず、高速処理が可能となるため、特に通信機器、通信機能を具備した電子計算機への応用に適する。

【0042】また、イベント情報ブロックの再利用が可能なことにより、イベント情報ブロックの領域確保、領域開放処理および送受信データのコピーの発生頻度を減少させることも、処理の高速化につながる。

【0043】さらに、イベント情報ブロックの内部構成を統一し、各キューに属するイベントを保持するメモリを共通領域から取得することにより、複数のイベントキューを持つために生じる、処理の複雑化を避けることが可能となる。

【図面の簡単な説明】

【図1】本発明キュー処理方式のキューの構成方法を示した図である。

【図2】本発明キュー処理方式におけるイベントのフォーマット図である。

【図3】本発明キュー処理方式のキュー及びイベントを用いた具体的な実施例を示した図である。

【図4】従来のキューの構成方法の例を示した図である。

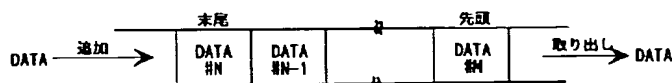
【図5】従来のキューの構成方法の例を示した図である。

【図6】図5において割り込みを禁止せずに、キューの操作を行った場合の例を示した図である。

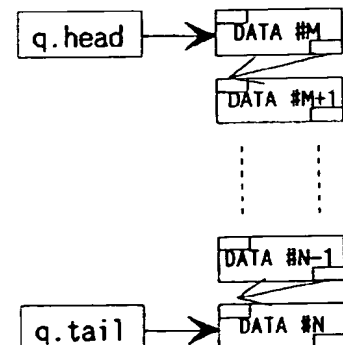
【符号の説明】

- 1～3 各イベントキュー
- 4～7 各イベント情報ブロック領域
- 8～10 各イベント
- 11 イベントキュー
- 12 スケジューラ
- 13～15 各イベント処理部
- 16 送受信処理モジュール

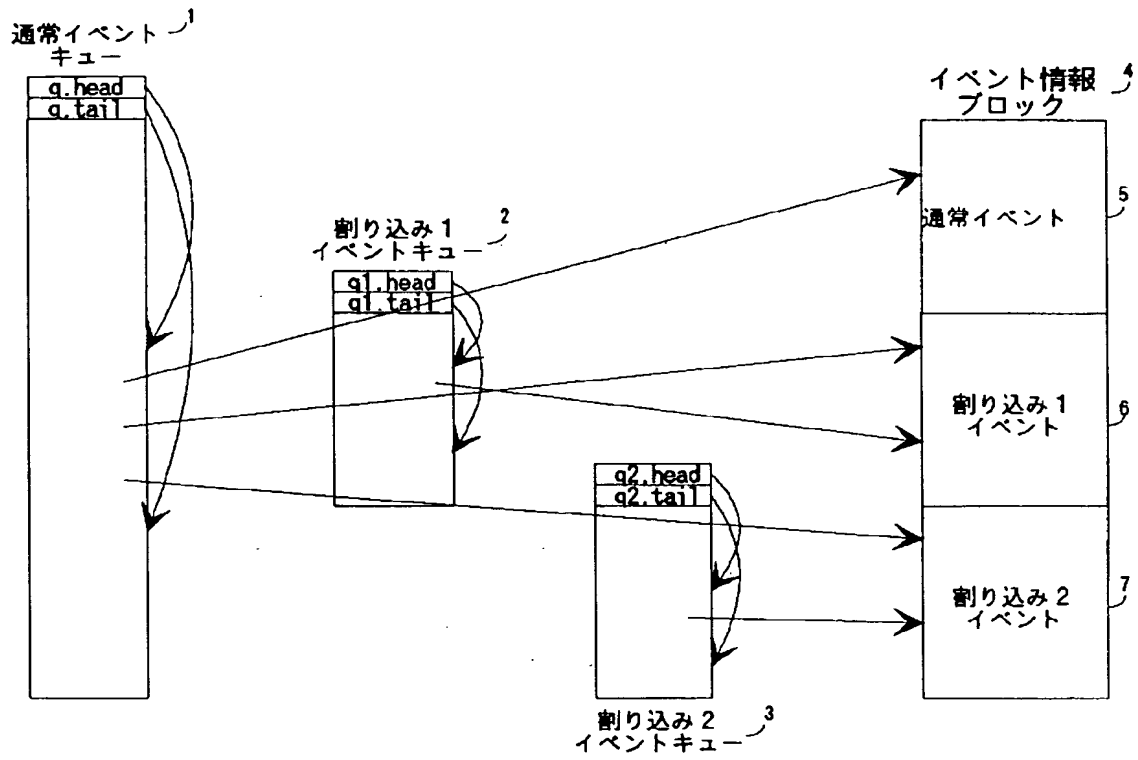
【図4】



【図5】



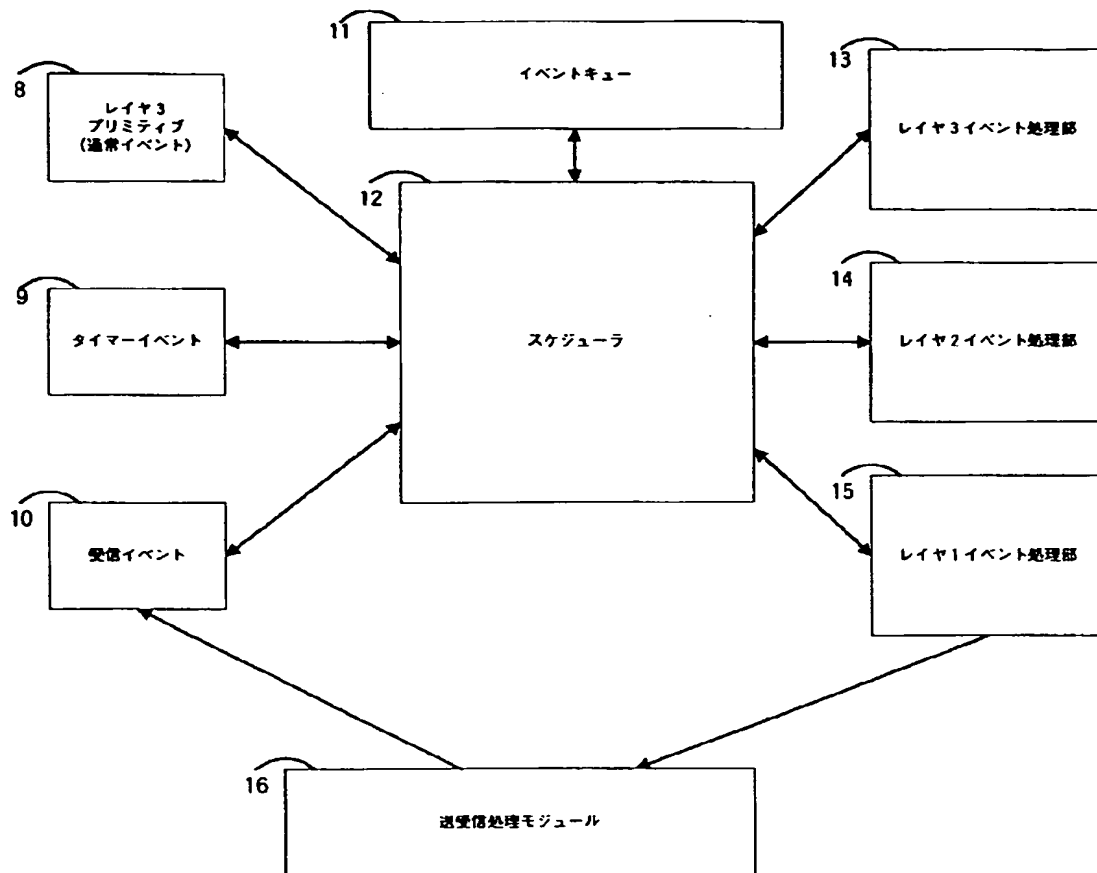
【図1】



【図2】

送信先
送信元
イベント種別
パラメータ長
パラメータ

【図3】



【図6】

